

Re: encryption/decryption help

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2005-01/2055.html>

From: Paul Rubin (//phr.cx_at_NOSPAM.invalid)

Date: 01/12/05

Date: 12 Jan 2005 12:35:45 -0800

"drs" <drs@remove-to-send-mail-ecpssoftware.com> writes:

> *Hi, I need to send secure data over an insecure network. To that end, I am
> needing to encrypt serialized data and then decrypt it. Is there a builtin
> way to do this in Python? MD5, SHA, etc encrypt, but I am not seeing a way
> to get back my data.*

No, Python doesn't include any reversible encryption functions, because of regulatory obstacles in some countries. Here's a function based on SHA:

<http://www.nightsong.com/phr/crypto/p3.py>

It's not ideal and it's nonstandard, but it's written in pure Python and still has reasonable performance and should have ok security.

It works on 32-bit processors but a simple fix is needed to make it work on 64-bit processors. I'll put that in when I get a chance.

> *Encryption is totally new to me, so any pointers of what to read up
> on would be appreciated.*

Rule #1 is that there are a lot of subtle mistakes that can kill you. Try to use standard solutions when you can, instead of doing anything ad-hoc.

The standard reference about crypto implementation is "Applied Cryptography" by Bruce Schneier. That's got all kinds of stuff about algorithms and protocols. You could also look at "Practical Cryptography" by Bruce Schneier and Niels Ferguson. That is more about what kinds of precautions you should take when implementing crypto. I disagree with some of what it says, but it's a start.

Also, anyone implementing any type of security system (crypto or not) should read "Security Engineering" by Ross Anderson.

> *As a side note, I understand that I could use https, but this would involve
> changing things that I may not be at liberty to change -- though if this
> turns out to be the best solution, then I'll find a way to use it.*

comp.lang.python: Re: encryption/decryption help

Using https is almost certainly a better solution than rolling up something yourself. Do it if the option is available to you.